Lecture 2

- Linearization

- Stability definitions

- Simulation in Matlab/Simulink

Material

- Glad& Ljung Ch. 11, 12.1,
  (Khalil Ch 2.3, part of 4.1, and 4.3)
  (Slotine and Li, pp 40-57)
- Lecture slides

Today's Goal

To be able to

- *linearize, both around equilibria and trajectories,*
- *explain definitions of local and global stability,*
- *check local stability and local controllability at equilibria*
- *simulate in Simulink,*

Example - Linearization around equilibrium point

The linearization of

$$\ddot{x}(t) = \frac{g}{l} \sin x(t)$$

around the equilibrium $x_0 = n\pi$ is given by

$$\ddot{\tilde{x}}(t) = \frac{g}{l} \sin(n\pi + \tilde{x}(t)) \approx \frac{g}{l}(-1)^n \tilde{x}(t)$$

Hint: $\sin(\alpha + \beta) = \sin(\alpha)\cos(\beta) + \sin(\beta)\cos(\alpha)$

2 minute exercise: Linearize

$$\ddot{x} + \dot{x}^3 - \dot{x}^2 - x = u$$

around the solution

$$x_0(t) = e^t, \qquad u_0(t) = e^{3t} - e^{2t}$$

Hint: First check if (x_0, u_0) is a solution. Then plug-in
$x(t) = e^t + \tilde{x}(t)$, $u(t) = e^{3t} - e^{2t} + \tilde{u}(t)$, expand the expressions,
and finally remove higher order terms (≥ 2) of $\tilde{x}$ and $\tilde{u}$.

Linearization Around a Trajectory

Idea:
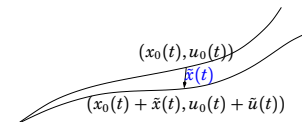 Make Taylor-expansion around a known solution $\{x_0(t), u_0(t)\}$

Neglect small terms (*i.e.*, keep the linear terms, as these will locally
dominate over the higher order terms).

Let

$$dx_0/dt = f(x_0(t), u_0(t)) \quad \text{be a known solution}$$

How will a small deviation $\{\tilde{x}, \tilde{u}\}$ from this solution behave?

$$\frac{d(x_0 + \tilde{x})}{dt} = f(x_0(t) + \tilde{x}(t), u_0(t) + \tilde{u}(t))$$

Linearization, explicit form

Consider $\dot{x}(t) = f(x(t), u(t))$ and assume $\dot{x}_0 = f(x_0(t), u_0(t))$

The linearization around $(x_0(t), u_0(t))$ is given by

$$\frac{d}{dt}(x(t) - x_0(t)) = A(t) \cdot (x(t) - x_0(t)) + B(t) \cdot (u(t) - u_0(t))$$

where (if dim x = dim u = 2)

$$
\begin{aligned}
A(t) &= \left.\frac{\partial f}{\partial x}\right|_{(x_0, u_0)} = \left.\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix}\right|_{(x_0(t), u_0(t))} \\
B(t) &= \left.\frac{\partial f}{\partial u}\right|_{(x_0, u_0)} = \left.\begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} \end{bmatrix}\right|_{(x_0(t), u_0(t))}
\end{aligned}
$$

Linearization, cont'd

The linearization of the output equation

$$y(t) = h(x(t), u(t))$$

around the nominal output $y_0(t) = h(x_0(t), u_0(t))$ is given by

$$(y(t) - y_0(t)) = C(t)(x(t) - x_0(t)) + D(t)(u(t) - u_0(t))$$

where (if dim y = dim x = dim u = 2)

$$
\begin{aligned}
C(t) &= \left.\frac{\partial h}{\partial x}\right|_{(x_0, u_0)} = \left.\begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} \end{bmatrix}\right|_{(x_0(t), u_0(t))} \\
D(t) &= \left.\frac{\partial h}{\partial u}\right|_{(x_0, u_0)} = \left.\begin{bmatrix} \frac{\partial h_1}{\partial u_1} & \frac{\partial h_1}{\partial u_2} \\ \frac{\partial h_2}{\partial u_1} & \frac{\partial h_2}{\partial u_2} \end{bmatrix}\right|_{(x_0(t), u_0(t))}
\end{aligned}
$$

Linearization Around a Trajectory, cont.

Let $(x_0(t), u_0(t))$ denote a solution to $\dot{x} = f(x, u)$ and consider
another solution $(x(t), u(t)) = (x_0(t) + \tilde{x}(t), u_0(t) + \tilde{u}(t))$:

$$
\begin{aligned}
\dot{x}(t) &= f(x_0(t) + \tilde{x}(t), u_0(t) + \tilde{u}(t)) \\
&= f(x_0(t), u_0(t)) + \frac{\partial f}{\partial x}(x_0(t), u_0(t))\tilde{x}(t) \\
&\quad + \frac{\partial f}{\partial u}(x_0(t), u_0(t))\tilde{u}(t) + O(\|\tilde{x}, \tilde{u}\|^2)
\end{aligned}
$$
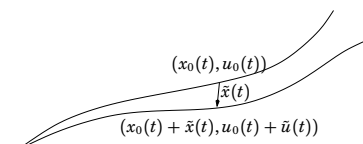
State-space form

Hence, for small $(\tilde{x}, \tilde{u})$, approximately

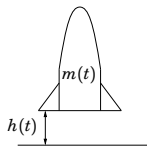$$\dot{\tilde{x}}(t) = A(x_0(t), u_0(t))\tilde{x}(t) + B(x_0(t), u_0(t))\tilde{u}(t)$$

where (if dim $x = 2$, dim $u = 1$)

$$A(x_0(t), u_0(t)) = \frac{\partial f}{\partial x}(x_0(t), u_0(t)) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix}\Big|_{(x_0(t), u_0(t))}$$

$$B(x_0(t), u_0(t)) = \frac{\partial f}{\partial u}(x_0(t), u_0(t)) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} \\ \frac{\partial f_2}{\partial u_1} \end{bmatrix}\Big|_{(x_0(t), u_0(t))}$$

Note that A and B are **time dependent**! However, if we don't linearize around a trajectory but linearize around an equilibrium point $(x_0(t), u_0(t)) \equiv (x_0, u_0)$ then A and B are constant.
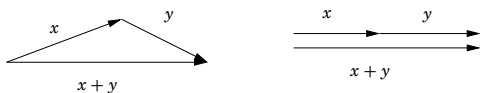
Example: Rocket

$$\dot{h}(t) = v(t)$$
$$\dot{v}(t) = -g + \frac{v_e u(t)}{m(t)}$$
$$\dot{m}(t) = -u(t)$$

Let $u_0(t) \equiv u_0 > 0$; $\quad x_0(t) = \begin{bmatrix} h_0(t) \\ v_0(t) \\ m_0(t) \end{bmatrix}$; $\quad m_0(t) = m_0 - u_0 t$.

Linearization: $\quad \dot{\tilde{x}}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{-v_e u_0}{m_0(t)^2} \\ 0 & 0 & 0 \end{bmatrix} \tilde{x}(t) + \begin{bmatrix} 0 \\ \frac{v_e}{m_0(t)} \\ -1 \end{bmatrix} \tilde{u}(t)$
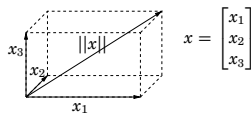
Definition: A *norm function* $\|x\| : x \in R^n \to R$ satisfies the following three properties:

▶ $\|x\| = 0$ if and only if (iff) $x = 0$, $\quad \|x\| > 0$ otherwise.
▶ $\|ax\| = a\|x\|$, for any positive a and any signal vector x.
▶ (The triangle inequality) $\|x + y\| \leq \|x\| + \|y\|$

Definition: (Euclidean norm)

$$\|x\| = (x_1^2 + x_2^2 + \ldots + x_n^2)^{1/2}$$

Asymptotic Stability

Definition The equilibrium $x = 0$ is **locally asymptotically stable (LAS)** if it

1) is stable
2) there exists $r > 0$ so that if $\|x(0)\| < r$ then

$$x(t) \longrightarrow 0 \text{ as } t \longrightarrow \infty.$$

(PhD-exercise: Show that 1) does not follow from 2))

Linearization, cont'd

The linearization of the output equation

$$y(t) = h(x(t), u(t))$$

around the nominal output $y_0(t) = h(x_0(t), u_0(t))$ is given by

$$(y(t) - y_0(t)) = C(t)(x(t) - x_0(t)) + D(t)(u(t) - u_0(t))$$

where (if dim $y = $ dim $x = 2$, dim $u = 1$)

$$C(t) = \frac{\partial h}{\partial x}\Big|_{(x_0, u_0)} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} \end{bmatrix}\Big|_{(x_0(t), u_0(t))}$$

$$D(t) = \frac{\partial h}{\partial u}\Big|_{(x_0, u_0)} = \begin{bmatrix} \frac{\partial h_1}{\partial u_1} \\ \frac{\partial h_2}{\partial u_1} \end{bmatrix}\Big|_{(x_0(t), u_0(t))}$$
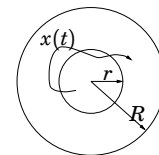
Part II: Stability definitions

Local Stability

Consider $\dot{x} = f(x)$ where $f(0) = 0$

Definition The equilibrium $x = 0$ is **stable** if, for any $R > 0$, there exists $r > 0$, such that

$$\|x(0)\| < r \quad \implies \quad \|x(t)\| < R, \quad \text{for all } t \geq 0$$

Otherwise the equilibrium point is **unstable**.

Global Asymptotic Stability

Definition The equilibrium is said to be **globally asymptotically stable (GAS)** if it is LAS and for all $x(0)$ one has

$$x(t) \to 0 \text{ as } t \to \infty.$$

Part III: Check local stability and controllability

Lyapunov's Linearization Method

Theorem Assume

$$\dot{x} = f(x)$$

has the linearization

$$\frac{d}{dt}(x(t) - x_0) = A(x(t) - x_0)$$

around the equilibrium point x_0 and put

$$\alpha(A) = \max \operatorname{Re}(\lambda(A))$$

- ► If $\alpha(A) < 0$, then $\dot{x} = f(x)$ is LAS at x_0,
- ► If $\alpha(A) > 0$, then $\dot{x} = f(x)$ is unstable at x_0,
- ► If $\alpha(A) = 0$, then no conclusion can be drawn.

(Proof in Lecture 4)

Example

The linearization of

$$\begin{aligned} \dot{x}_1 &= -x_1^2 + x_1 + \sin(x_2) \\ \dot{x}_2 &= \cos(x_2) - x_1^3 - 5x_2 \end{aligned}$$

at $x_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ gives $A = \begin{bmatrix} -1 & 1 \\ -3 & -5 \end{bmatrix}$

Eigenvalues are given by the *characteristic equation*

$$0 = \det(\lambda I - A) = (\lambda + 1)(\lambda + 5) + 3$$

This gives $\lambda = \{-2, -4\}$, which are both in the left half-plane, hence the *nonlinear system* is LAS around x_0.

Local Controllability

Theorem Assume

$$\dot{x} = f(x, u)$$

has the linearization

$$\frac{d}{dt}(x(t) - x_0) = A(x(t) - x_0) + B(u(t) - u_0)$$

around the equilibrium (x_0, u_0) then

- ► (A, B) controllable $\Rightarrow$ $f(x, u)$ nonlinear locally controllable

Here **nonlinear locally controllable** is defined as:

For every $T > 0$ and $\varepsilon > 0$ the set of states $x(T)$ that can be reached from $x(0) = x_0$, by using controls satisfying $\|u(t) - u_0\| < \varepsilon$, contains a small ball around x_0.

5 minute exercise:

Is the ball and beam

$$\frac{7}{5}\ddot{x} = x\dot{\phi}^2 + g\sin\phi + \frac{2r}{5}\ddot{\phi}$$

nonlinearly locally controllable around $\dot{\phi} = \phi = x = \dot{x} = 0$ (with $\ddot{\phi}$ as input)?

Remark: This is a bit bit more detailed model of the ball and beam than we saw in Lecture 1.

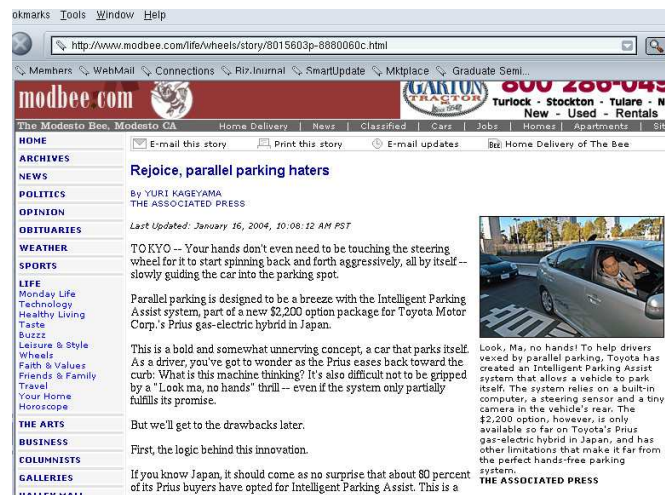## Example

An inverted pendulum with vertically moving pivot point

$$\ddot{\phi}(t) = \frac{1}{l}\left(g + u(t)\right)\sin(\phi(t)),$$

where $u(t)$ is acceleration, can be written as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{l}(g + u)\sin(x_1) \end{aligned}$$

However...

Bosch 2008 (Automatic parking assistance)

- ► Multiple turns
- ► parking lot > car length + 80 cm

More parking in lecture 12

Example, cont.

The linearization around $x_1 = x_2 = 0, u = 0$ is given by

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = \frac{g}{l}x_1$$

It is not controllable, hence no conclusion can be drawn about nonlinear controllability

However, simulations show that the system is stabilized by

$$u(t) = \varepsilon\omega^2 \sin(\omega t)$$

if ω is large enough !
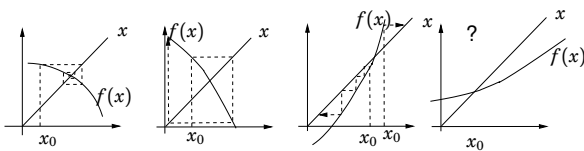
Demonstration We will come back to this example later.

Example (cont'd): Numerical iteration
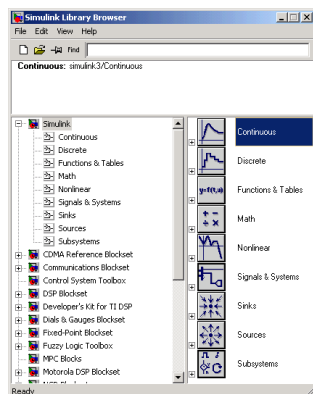
$$x_{k+1} = f(x_k)$$

to find fixed point

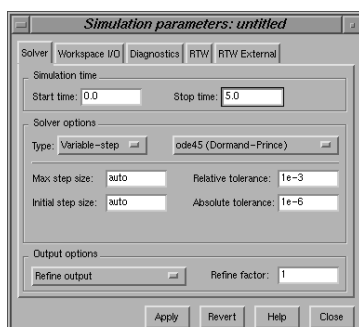$$x^* = f(x^*)$$

When does the iteration converge?

Simulink


```
> matlab
>> simulink
```

Bonus — Discrete Time

Many results are parallel (observability, controllability,...)

Example: The difference equation

$$x_{k+1} = f(x_k)$$

is asymptotically stable at x^* if the linearization

$\left.\dfrac{\partial f}{\partial x}\right|_{x^*}$ has all eigenvalues in $|\lambda| < 1$

(that is, within the unit circle).

Part IV: Simulation

Often the only method

$\dot{x} = f(x)$
- ACSL
- Simnon
- Simulink

$F(\dot{x}, x) = 0$
- Omsim
  http://www.control.lth.se/~cace/omsim.html
- Dymola http://www.dynasim.se/
- Modelica
  http://www.dynasim.se/Modelica/index.html

Special purpose
- Spice (electronics)
- EMTP (electromagnetic transients)
- Adams (mechanical systems)

Simulink, An Example

```
File -> New -> Model
Double click on Continuous
Transfer Fcn
Step (in Sources)
Scope (in Sinks)
Connect (mouse-left)
Simulation->Parameters
```
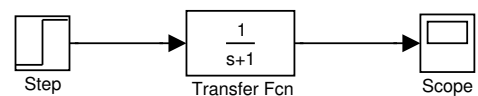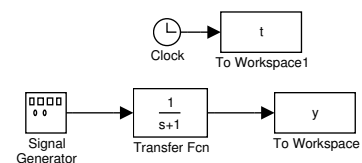

Choose Simulation Parameters

Don't forget "Apply"

Save Results to Workspace

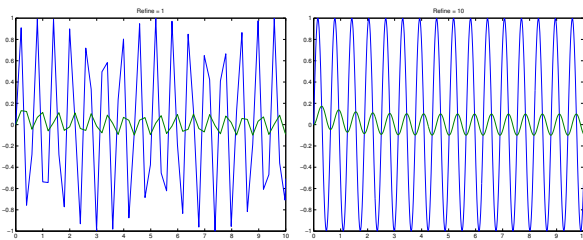Check "Save format" of output blocks ("Array" instead of "Structure")

```
>> plot(t,y)
```

(or use "Structure" which also contains the time information.)

How To Get Better Accuracy

Modify Refine, Absolute and Relative Tolerances, Integration method
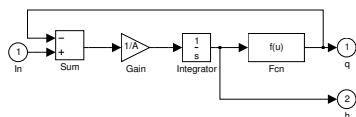
Refine adds interpolation points:

Submodels, Example: Water tanks
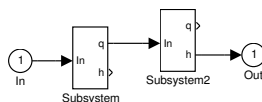
Equation for one water tank:

$$\dot{h} = (u - q)/A$$
$$q = a\sqrt{2g}\sqrt{h}$$

Corresponding Simulink model:

Make a subsystem and connect two water tanks in series.

Linearization in Simulink, cont.

Use the command `linmod` to find a linear approximation of the system around an operating point:

```
>> [aa,bb,cc,dd]=linmod('flow',x0,u0);
>> sys=ss(aa,bb,cc,dd);
>> bode(sys)
```

Computer exercise

Simulation of JAS 39 Gripen

- ▶ Simulation
- ▶ Analysis of PIO using describing functions
- ▶ Improve design

Use Scripts to Document Simulations

If the block-diagram is saved to `stepmodel.mdl`,
the following Script-file `simstepmodel.m` simulates the system:

```
open_system('stepmodel')
set_param('stepmodel','RelTol','1e-3')
set_param('stepmodel','AbsTol','1e-6')
set_param('stepmodel','Refine','1')
tic
sim('stepmodel',6)
toc
subplot(2,1,1),plot(t,y),title('y')
subplot(2,1,2),plot(t,u),title('u')
```
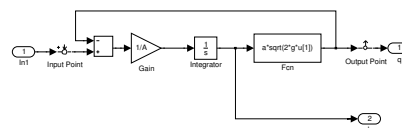
Linearization in Simulink

Use the command `trim` to find *e.g.,* stationary points to a system `>> A=2.7e-3;a=7e-6,g=9.8;`

```
>> % Example to find input u for desired states/output
>> [x0,u0,y0]=trim('flow',[0.1 0.1]',[],0.1)
x0 =
    0.1000
    0.1000
u0 =
    8.3996e-06
y0 =
    0.1000
```

Linearization in Simulink; Alternative

By right-clicking on a signal connector in a Simulink model you can add "Linearization points" (inputs and/or outputs).
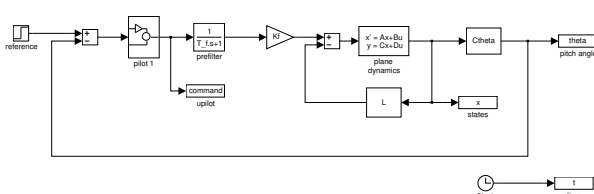

Start a "Control and Estimation Tool Manager" to get a linearized model by
`Tools -> Control Design ->Linear analysis ...`

where you can set the operating points, export linearized model to Workspace (Model-> Export to Workspace) and much more.

Today

- ▶ Linearization, both around equilibria and trajectories,
- ▶ Definitions of local and global stability,
- ▶ How to check local stability and local controllability at equilibria
- ▶ Simulation tool: Simulink,

Next Lecture

- ▶ Phase plane analysis
- ▶ Classification of equilibria